# Zig Zag Sequence

In this challenge, the task is to debug the existing code to successfully execute all provided test files.

Given an array of n distinct integers, transform the array into a zig zag sequence by permuting the array elements. A sequence will be called a zig zag sequence if the first k elements in the sequence are in increasing order and the last k elements are in decreasing order, where k = (n + 1)/2. You need to find the *lexicographically smallest* zig zag sequence of the given array.

# Example.

 $a = \left[2, 3, 5, 1, 4\right]$ 

Now if we permute the array as [1, 4, 5, 3, 2], the result is a zig zag sequence.

Debug the given function **findZigZagSequence** to return the appropriate zig zag sequence for the given input array.

**Note:** You can modify at most *three* lines in the given code. You cannot add or remove lines of code.

To restore the original code, click on the icon to the right of the language selector.

### **Input Format**

The first line contains t the number of test cases. The first line of each test case contains an integer n, denoting the number of array elements. The next line of the test case contains n elements of array a.

# Constraints

 $egin{aligned} 1 \leq t \leq 20 \ 1 \leq n \leq 10000 \ (n ext{ is always odd}) \ 1 \leq a_i \leq 10^9 \end{aligned}$ 

# **Output Format**

For each test cases, print the elements of the transformed zig zag sequence in a single line.